

	Type	L #	Hits	Search Text	DBs	Time Stamp
1	BRS	L1	3	4837738.pn. or 4851989.pn. or 5408626.pn.	USP AT; US-P GPU B; EPO; JPO; IBM_ TDB	2004/07/0 7 10:59
2	BRS	L2	1201	carry adj5 look adj5 ahead	USP AT; US-P GPU B; EPO; JPO; IBM_ TDB	2004/07/0 7 11:00
3	BRS	L3	68	(carry adj5 look adj5 ahead).ti.	USP AT; US-P GPU B; EPO; JPO; IBM_ TDB	2004/07/0 7 11:43
4	BRS	L4	1	"20020188642"	USP AT; US-P GPU B; EPO; JPO; IBM_ TDB	2004/07/0 7 11:44

	Type	L #	Hits	Search Text	DBs	Time Stamp
5	BRS	L5	313	generate with propagate and 2	USP AT; US-P GPU B; EPO; JPO; IBM_ TDB	2004/07/07 11:45
6	BRS	L6	47	generat\$4 with propagat\$4 with combin\$4 and 2	USP AT; US-P GPU B; EPO; JPO; IBM_ TDB	2004/07/07 11:46
7	BRS	L7	39	generat\$4 with propagat\$4 with combin\$4 with carry and 2	USP AT; US-P GPU B; EPO; JPO; IBM_ TDB	2004/07/07 11:46
8	BRS	L8	11	(generat\$4 with propagat\$4 with combin\$4 with carry).clm. and 2	USP AT; US-P GPU B; EPO; JPO; IBM_ TDB	2004/07/07 11:46

US-PAT-NO: 5122982

DOCUMENT-IDENTIFIER: US 5122982 A

TITLE: Carry generation method and apparatus

DATE-ISSUED: June 16, 1992

US-CL-CURRENT: 708/710

APPL-NO: 07/ 619741

DATE FILED: October 17, 1990

PARENT-CASE:

This is a continuation of copending application Ser. No. 07/434,467 filed on Nov. 13, 1989, now abandoned, which is a continuation of application Ser. No. 07/162,081 filed Feb. 29, 1988, now abandoned.

----- KWIC -----

Abstract Text - ABTX (1):

Methods and apparatus for digital addition are disclosed in particular methods and apparatus for "carrying" in digital addition, and methods for designed carry circuits using circle and tally diagrams. A general method for

creating and analyzing carry circuits is developed which not only permits the analysis of existing carry circuits such as those of the ripple-carry adder and the carry look-ahead adder; but also leads to new faster carry circuits.

Brief Summary Text - BSTX (9):

One technique which has been employed to expedite addition separates the function of simple addition from that of the calculation of the carry bits.

Probably the most common example today is the carry look-ahead adder. The

standard carry look-ahead adder circuitry is obtained by considering the

Boolean functions that define addition and applying a little algebra to them

(see Computer System Architecture, first edition, 1976, M. Morris Mano,

Prentice-Hall, pages 242-249). In this process, two Boolean terms are usually

introduced, called generate and propagate. The carry look-ahead method of

addition has been standard for many years with only minor changes. (See The

TTL Data Book, volume 2, Texas Instruments, 1985, pages 3-721 to 3-726 and the

F100K ECL Data Book, Fairchild, 1982, pages 3-146 to 3-151).

Brief Summary Text - BSTX (12):

The approach taken by the present invention does not consider the Boolean

equations that define addition, but instead uses an intuitive

understanding of
the generate and propagate bits: what they mean, and ways they
can be combined
and generalized. A general method for creating and analyzing
carry circuits
was developed. This method not only permits the analysis of
existing carry
circuits such as those of the ripple-carry adder and the carry
look-ahead
adder; but also leads to new carry circuits, some of which are
faster than
carry look-ahead circuits.

Drawing Description Text - DRTX (15):

**FIG. 11 is a schematic diagram of a known four bit carry
look-ahead module.**

Drawing Description Text - DRTX (16):

**FIG. 12 is a schematic block diagram of a 16 bit carry look-ahead
circuit
using the modules of FIG. 11.**

Drawing Description Text - DRTX (17):

**FIGS. 13A and 13B are circle and tally diagrams for a carry
look-ahead
circuit, as discussed in Analysis B.**

Detailed Description Text - DETX (48):

**FIG. 8 shows the simplified four bit group of FIG. 7 in the new
tally
diagram notation. This tally diagram contains all the information
necessary**

for producing an actual carry generation circuit. The tally diagram can be used to show some known circuits (carry look ahead and ripple carry) as well as all of the circuits of our invention. The following rules describe how circuits are represented by the tally diagrams and how their characteristics can be examined.

Detailed Description Text - DETX (67):

The method of the present invention can be used to analyze carry look-ahead circuits of the known type. Since the carry look-ahead circuit is not characterized by a simple expression as is the ripple-carry adder of Analysis A, a particular instance of the carry look-ahead adder is considered. This example is the carry circuit for adding two 16 bit numbers. Further, in this example, the fan-in is limited to four, so no interval is formed from more than four other intervals and the fan-out is also limited to four, so no interval is used to form more than four intervals in the next stage. FIG. 11 is a schematic diagram of a typical carry look-ahead generator circuit 1000. This circuit typically is used repeatedly in a modular fashion to form carry circuits of any desired size. FIG. 12 shows how modules 1101-1105, like circuit 1000, can be used to form a 16 bit adder 1100. While at first glance

FIG. 12 appears to depict two levels or stages, there are in fact three stages because signals are returned on 1111-1113 by module 1105 of the second stage to three modules 1101-1103 of the first stage. In FIG. 12, the lines labeled as carries in FIG. 11 have been relabeled as generates for intervals to obtain for this adder the equations those developed above. The circle diagram for this carry circuit is shown in FIG. 13A and the corresponding tally diagram is shown in FIG. 13B. The equations are set forth below:

Detailed Description Text - DETX (73):

This example is a new carry circuit that is faster than the carry look-ahead circuit of Analysis B for addition of numbers more than eight bits wide. This example has the same fan-in and fan-out restrictions as the known carry look-ahead of Analysis B circuit and thus is a practical circuit for addition of numbers more than several bits wide. The general form for each stage can be determined from the circle diagram of FIG. 14A and the tally diagram of FIG. 14B, and the following equations for this example can be determined from inspection of either of these diagrams:

Detailed Description Text - DETX (77):

Notice that this new carry circuit only requires two stages for addition of

two 16 bit numbers and a fan-in and fan-out of 4; whereas, the known carry look-ahead circuit for addition of the same numbers and having the same fan-in and fan-out, analyzed in Analysis B above, required three stages. In general, where N is the number of bits to be added and b is the maximum fan-in permitted, the carry look-ahead needs $2\lceil \log_{\text{sub}.b} N \rceil - 1$ stages while the new carry circuit of this example needs only $\lceil \log_{\text{sub}.b} N \rceil$ stages, where "ceil[x]" is the integer ceiling. That is ceil[x] is the smallest integer not less than x . V. M. Khrapchenko, "Asymtotic Estimate of Addition Time of a Parallel Adder," in Systems Theory Research (A. A. Lyapunor, ed. 1967) (English trans. 1970) 105 has shown that this is the smallest number of possible stages; however, that paper did not provide any practical example of how the minimum number of stages might be accomplished.

Detailed Description Text - DETX (82):

Note that, a comparison of the $2\lceil \log_{\text{sub}.b} N \rceil - 1$ limit of the carry look-head carry circuit with the $\lceil \log_{\text{sub}.b} N \rceil$ limit of my new fast carry circuit of this example indicates that the speed advantage of the new carry circuit of this example continues to improve as the width of the numbers to be added increases. For example, the carry circuit in accordance with this

Example 2 requires only three stages for the addition of two 64 bit wide numbers, whereas the known carry look-ahead circuit analyzed in Analysis B and having the same fan-in and fan-out would require 5 stages.

Detailed Description Text - DETX (83):

There is, of course, a trade-off between the speed provided and space occupied by carry circuits. For example, the new carry circuits of this Example 4 require many more gates and wires than the carry look-ahead circuit does. As higher levels of circuit integration are attained and other progress is made in semiconductor technology; however, this is not a serious handicap.

Detailed Description Text - DETX (94):

In Examples 6 and 7, shown in FIGS. 19 and 20, respectively, two other new carry circuits have been designed by use of tally diagrams to demonstrate the utility of this method and the range of carry circuits which can be designed in accordance with my invention. A random number generator was used to generate a series of numbers within a desired range of fan-in (four or less in these examples). This series of numbers was used to determine the number of dots to place on each of the linear segments of these tally diagrams. The resulting diagrams are shown in FIGS. 19 and 20. Notice that the carry

circuit of FIG.

20, generated in this way, is as fast as the carry look-ahead circuit of

Analysis B and is on the same order of complexity.

Detailed Description Text - DETX (98):

Another way to handle the carry-in is to use an $N+1$ column carry circuit for

adding N bit numbers instead of an N column carry circuit. The two numbers to

be added are positioned in the left N columns (positions 1 through N) and the

carry-in is positioned in the right most column (position 0). Then for $k > 0$,

define $G_{\text{sub}.k}$ in the usual way, but for $k=0$, define $G_{\text{sub}.0} = C_{\text{sub}.0}$.

Depending on the value of N and the structure of the carry circuit considered,

it may not be slower to use an $N+1$ bit carry circuit than an N bit carry

circuit. This is how the carry look-ahead circuits handle the carry-in.

Detailed Description Text - DETX (99):

The method and apparatus described here includes many new carry circuits

which are both practical to implement in VLSI circuits and are up to nearly

twice as fast as carry look-ahead circuits. The fastest of the new carry

circuits are faster than any carry circuit I am aware of and achieve the

theoretical bound for the fastest possible adder. My method makes it very easy

to create new adders, each with its own properties. Thus it is now easy to construct an add circuit with properties appropriate for any given situation.

Claims Text - CLTX (19):

18. A method of operation of a digital adder for the digital addition of two numbers A and B to produce a sum S comprising the steps of calculating the carry bits C.sub.k by use of a plurality of logic cells, where k is the column position within the numbers being added and C.sub.k is the carry bit from the preceding column, and adding A.sub.k, B.sub.k and C.sub.k, the improvement comprising calculation of the carry bits C.sub.k in a sequence of not more than $2\text{ceil}(\log_{\text{sub}.b} N)-2$ successive stages, where b is the maximum number of columns of information to be combined in a single logic cell, b is greater than 2, N is the total number of columns, and in each stage, carry-generate and carry-propagate signals are produced substantially within the same time period for all columns for which the carry bit into the next column has not yet been determined.

Claims Text - CLTX (22):

21. An N bit carry generate circuit for use in an adder having a first circuit for receiving two digital signals A and B and producing

generate and propagate signals, a carry generate circuit, and a combinatorial circuit for receiving the propagate signals from the first circuit and carry signals from the carry generate circuit and for producing the sum of A and B, said carry generate circuit comprising a plurality of combining logic cells arranged in successive stages, each of said logic cells receiving at least two and not more than b generate signals, where b is the fan-in of a logic cell, and not more than b propagate signals, producing a combined generate signal from all of said logic cells; producing a combined propagate signal from at least some of said logic cells; wherein the maximum number of combining logic cells in any signal path is less than or equal to $2^{\lceil \log_{\text{sub}.b} N \rceil - 2}$.

Claims Text - CLTX (24):

23. The carry generate circuit of claim 22 wherein at each column position of a stage after the first stage, b carry-generate signals and b carry-propagate signals are combined by a logic cell unless the carry-generate signal $G_{\text{sub}.k,0}$ is produced in that or a preceding stage.

Claims Text - CLTX (26):

25. The carry generate circuit of claim 24 wherein at each column position of a stage after the first stage b carry-generate signals and b carry-propagate

signals are combined by a logic cell unless the carry-generate signal G.sub.k,0 is produced in that or a preceding stage.

Claims Text - CLTX (28):

27. The carry generate circuit of claim 20 wherein at each column position of a stage after the first stage b carry-generate signals and b carry-propagate signals are combined by a logic cell unless the carry-generate G.sub.k,0 is produced in that or a preceding stage.

Claims Text - CLTX (32):

31. The carry generate circuit of claim 21 wherein at each column position of a stage after the first stage b carry-generate signals and b carry-propagate signals are combined by a logic cell unless the carry-generate signal G.sub.k,0 is produced in that or a preceding stage.

Other Reference Publication - OREF (5):

Sinha & Srimani, "Fast Implementation of Group Carry Look-Ahead in a CMOS Adder", IEEE Transactions on Computers, vol. 38, No. 3, Mar. 1989, pp. 424-431.

Other Reference Publication - OREF (6):

"Fast Implementation of Group Carry Look-Ahead in a CMOS Adder", IBM Tech. Disclosure Bulletin, vol. 29, No. 4, Sep. 1986, pp. 1842-1845.

Other Reference Publication - OREF (7):

Weinberger, "Improved Carry-Look-Ahead", IBM Tech. Disclosure Bulletin, vol. 21, No. 6, Nov. 1978, pp. 2460-2461.

US-PAT-NO: 5278783

DOCUMENT-IDENTIFIER: US 5278783 A

TITLE: Fast area-efficient multi-bit binary adder with low fan-out signals

DATE-ISSUED: January 11, 1994

US-CL-CURRENT: 708/711

APPL-NO: 07/ 969124

DATE FILED: October 30, 1992

----- KWIC -----

Abstract Text - ABTX (1):

A carry look-ahead adder obtains high speed with minimum gate fan-in and a regular array of area-efficient logic cells in a datapath by including a first row of propagate-generate bit cells, a second row of block-propagate bit cells generating a hierarchy of block-propagate and block-generate bits, a third row of carry bit cells; and a bottom level of sum bit cells. The second row of block-propagate bit cells supply the block-propagate and block-generate bits to the first carry bit cells in chained segments of carry bit cells. In a

preferred embodiment for a 32-bit complementary metal-oxide semiconductor (CMOS) adder, the logic gates are limited to a fan-in of three, and the block-propagate bit cells in the second row are interconnected to form two binary trees, each including fifteen cells, and the carry cells are chained in segments including up to four cells. In general, the interconnections between the block-propagate bit cells are derived from a graph which is optimized to meet the constraints of fast static complementary metal-oxide-semiconductor (CMOS) circuit design: low fan-out and small capacitance load on most signals. Sufficient gain stages are present in the binary trees to build-up to a large drive capability where the large drive capability is needed.

Brief Summary Text - BSTX (3):

The present invention relates generally to a and more particularly to a multi-level carry look-ahead adder. The invention specifically relates to a multi-level carry lookahead adder implemented as an array of regularly-spaced rows and columns of logic cells in a datapath.

Brief Summary Text - BSTX (12):

One disadvantage of the adder circuit 20 is that the speed of the adder is limited by the time for a carry signal to propagate left-to-right through the

chain of carry bit cells 25, 26 from the carry input $C_{sub.-1}$ to the carry output $C_{sub.n-1}$. In particular, the carry propagation time is a linear function of the number of columns n in the adder, and therefore the adder 20 is very slow when it has a large number n of columns or bits. A known solution to this problem is to use carry look-ahead logic to reduce the time for generating the more significant carry bits. The carry look-ahead logic has logic gates for more directly solving the carry function:

Brief Summary Text - BSTX (14):

As disclosed in Kai Huang, Computer Arithmetic, John Wiley & Sons, New York, N.Y., 1979, pp. 84-90, the carry function can be computed by "block carry generate" G^* and "block carry propagate" P^* functions in multi-level circuits.

Shown in FIG. 3.13 on page 90 of Huang, for example, is a two-level carry look-ahead adder with a 32-bit word length arranged in an 8-by-4 configuration.

The carry generation logic includes an upper level of eight four-bit block-carry look-ahead units and a lower level having an 8-bit carry look-ahead unit. Each four-bit block-carry look-ahead unit generates block carry generate and block carry propagate functions, for $i=3, 7, 11, 15, 19, 23, 27$, and 31:

Brief Summary Text - BSTX (18):

A carry-skip scheme is disclosed in Oklobdzija et al., "Some optimal schemes for ALU implementation in VLSI technology," Proceedings of the 7th Symposium on Computer Arithmetic, IEEE, Piscataway, N.J. (1985), pp. 2-8. The carry-generate portion, which consumes a large amount of logic, is eliminated.

As in a carry look-ahead adder, the bits to be added are divided into groups.

A circuit is provided for detecting when a carry signal entering a group will ripple through the group. When this condition is detected, the carry is allowed to skip over the group.

Brief Summary Text - BSTX (19):

Graph representations for designing area-time efficient VLSI adders are disclosed in Han et al., "Fast area-efficient VLSI adders," Proceedings of the 1987 Symposium on Computer Architecture, IEEE, Piscataway, N.J. (1987), pp.

49-56. When a prefix graph is used as a basis for designing binary addition

circuitry in VLSI, each node of the graph represents a set of logic equations.

Thus, each node can be thought of as a processing element that will be expanded

from being a point in the graph to occupy a fixed amount of area in the layout.

For binary addition, four types of processing elements can be used: pggen,

black, white, and sum. The pggen cell produces initial p and g signals (carry

propagation and generation signals). The black cell comprises a

pair of p signals and a pair of g signals to generate a p and g signal at a lower level. Two different types of black cells are used: a positive input, negative output cell; and a negative input, positive output cell. The white cell is a simple inverter that inverts a p signal and a g signal. The sum cell generates the sum bit from a propagate bit, a generate bit, and two carry bits. Because the carries produced by the carry generation circuitry alternate between being positive and negative, there are two types of sum cells: one type takes two carries without inversion, and the other takes two carries with inversion. The carry look-ahead adder based on the hybrid prefix algorithm is densely packed by using a folding method. The folding method places two levels of the prefix graph into one level of the layout, since space is available to embed cells.

Detailed Description Text - DETX (3):

The present invention concerns a high-speed multi-level carry look-ahead adder implemented as an array of regularly-spaced rows and columns of logic cells in a datapath. In particular, the present invention incorporates carry look-ahead logic into the basic adder configuration of FIG. 1 in such a way as to obtain the advantages of the basic configuration of FIG. 1 without the

disadvantage of low speed. Although a specific embodiment of a 32-bit adder incorporating the present invention will be described below with reference to FIGS. 12A, 12B, 12C, and 12D, it should be understood that the present invention is generally applicable to binary adders for adding numbers having a large number of bits. In any case, a binary adder, in accordance with the present invention, can be constructed from a number of primitive cells of logic gates. Some of these primitive cells correspond to the logic functions used in the conventional adder of FIG. 1, and other of the cells are used for the look-ahead carry logic which is not found in the conventional adder of FIG. 1.

Detailed Description Text - DETX (41):

In view of the above, there has been described a carry look-ahead adder that obtains high speed with minimum gate fan-in and a regular array of area-efficient logic cells in a datapath. The carry-look ahead logic uses a hierarch of recurrence solver cells to maximize speed. The recurrence solver cells are interconnected in two binary trees, so that the recurrence solver cells can be constructed of gates having a minimum fan-in. To minimize fan-out, the low-order sides of the trees are tapped at intervals to feed signals to segments of chained carry-bit cells. Although some of the taps of the trees

may have a relatively high fan-out, sufficient gate levels exist at these taps to build-up drive strength. Therefore, a very optimum architecture results for building fast binary adders from static CMOS gates.

Claims Text - CLTX (5):

wherein the carry bits which are generated but not computed directly from a corresponding generate bit $G_{sub.i}$ and a corresponding propagate bit $P_{sub.i}$ and a corresponding carry-in bit $C_{sub.i-1}$ are computed by generating a hierarchy of block-propagate and block-generate bits, including block-propagate bits and block-generate bits computed at a base level of said hierarchy by combining pairs of adjacent propagate bits $P_{sub.i+1}$, $P_{sub.i}$ and pairs of adjacent generate bits $G_{sub.i+1}$, $G_{sub.i}$; and

Claims Text - CLTX (9):

4. The method as claimed in claim 3, wherein said first one of said binary trees includes block-generate bits resulting from combining the carry-in $C_{sub.-1}$ with the lower-order propagate bits and generate bits so that the first one of said binary trees has a root which includes a carry bit $C_{sub.x}$ of order x of about one-half of n .

Claims Text - CLTX (20):

10. The adder as claimed in claim 9, wherein said first one of said binary trees combines the carry-in C.sub.-1 to the adder with the lower-order propagate bits and generate bits so that the first one of said binary trees has a root consisting of one of said block-generate cells that generates a carry bit C.sub.x of order x of about one-half of n.

Claims Text - CLTX (36):

21. The adder as claimed in claim 20, wherein said first one of said binary trees combines the carry-in C.sub.-1 to the adder with the lower-order propagate bits and generate bits so that the first one of said binary trees has a root consisting of one of said block-generate cells that generates a carry bit C.sub.x of order x of about one-half of n.

Claims Text - CLTX (50):

28. The adder as claimed in claim 25, wherein said first one of said binary trees combines the carry-in C.sub.-1 to the adder with the lower-order propagate bits and generate bits so that the first one of said binary trees has a root consisting of one of said block-generate cells that generates a carry bit C.sub.x of order x of about one-half of n.

US-PAT-NO: 4218747

DOCUMENT-IDENTIFIER: US 4218747 A

TITLE: Arithmetic and logic unit using basic cells

DATE-ISSUED: August 19, 1980

US-CL-CURRENT: 708/234, 257/E27.108 , 326/37 , 326/53

APPL-NO: 05/ 912451

DATE FILED: June 5, 1978

PARENT-CASE:

CROSS REFERENCE TO RELATED APPLICATION

CELLULAR INTEGRATED CIRCUIT AND HIERARCHICAL METHOD,
invented by Zasio, et
al., Ser. No. 847,478, filed Nov. 1, 1977 and assigned to the same
assignee
as the present invention.

----- KWIC -----

Abstract Text - ABTX (1):

**An arithmetic and logic unit (ALU) formed by a small number of
different
types of basic cells suitable for cellular integration to form large
scale**

integrated (LSI) semiconductor circuits. The arithmetic and logic unit is formed from a plurality of 1-bit ALU cells. Each ALU cell is formed by a plurality of cellular integrated basic cells where each ALU cell includes two data inputs, A and B, and responsively produces a data output, F. Each ALU cell has provision for some type of carry circuit, carry ripple or carry look-ahead.

In a carry ripple example, the 1-bit ALU cells are of two basic types, an odd type and an even type. The carry-out from an odd type cell is connected as a carry-in to an even type cell and similarly, the carry-out of an even type cell is connected as a carry-in to an odd type cell. The arithmetic and logic unit is formed by a plurality of alternating odd and even 1-bit ALU cells.

Detailed Description Text - DETX (26):

In FIG. 10, the carry ripple circuits 22 and 23 are one example of carry circuits which can be employed with the ALU cells of FIGS. 7 and 8. Since the carry circuits are independent of the other parts of the ALU cells, the carry circuits may be implemented in any other convenient and equivalent form, carry ripple or carry look-ahead.

Claims Text - CLTX (8):

7. The unit of claim 6 including a carry circuit formed of an INVERT basic

cell for inverting one of said bit propagate or said bit generate signals and including an AND-OR-INVERT basic cell for receiving a carry-in signal for combining the bit propagate, bit generate and carry-in signals as inverted to form a carry-out signal.

Claims Text - CLTX (13):

10. An arithmetic and logic unit comprising a plurality of 1-bit arithmetic and logic unit (ALU) cells where each ALU cell includes first and second data input and a data output and includes a carry-in and a carry-out and where each ALU cell includes a plurality of basic cells including means for forming a bit propagate signal, including means for forming a bit generate signal where said bit generate signal is of the opposite polarity of said bit propagate signal, including first means for logically combining said bit propagate and said bit generate signals to provide said data output, and including a carry circuit portion having an inverter for inverting one of said bit propagate or said bit generate signals and having second means for logically combining the carry propagate and carry generate signals as inverted and said carry-in to form said carry-out,

Claims Text - CLTX (26):

**16. The unit of claim 15 having odd type carry circuit portions wherein
said inverter inverts said carry generate signal G to form G and
wherein said
second means for logically combining combines a carry-in C.sub.in
together with
P and G to form a carry-out C.sub.out and having even type carry
circuit
portions wherein said inverter inverts the carry propagate P to P
and wherein
said second means for logically combining logically combines a
carry-in
C.sub.in together with P and G to form a carry-out C.sub.out.**

US-PAT-NO: 3993891

DOCUMENT-IDENTIFIER: US 3993891 A

****See image for Certificate of Correction****

**TITLE: High speed parallel digital adder employing
conditional
and look-ahead approaches**

DATE-ISSUED: November 23, 1976

US-CL-CURRENT: 708/710, 708/274 , 708/712 , 708/714

APPL-NO: 05/ 593167

DATE FILED: July 3, 1975

----- KWIC -----

Brief Summary Text - BSTX (9):

The carry propagate, carry generate and carry annihilate information provided by each of the conditional sum adder stages, as described above, are applied to carry look-ahead logic for simultaneously producing a look-ahead signal and a look-ahead carry-not signal for each stage. These look-ahead carry and carry-not signals are then applied to selection AND gates along with respective ones of the conditional sums for selecting the particular

**one of the
conditional sums of each stage which is to be used as the
corresponding output
digit of the sum of the applied addend and augend.**

Drawing Description Text - DRTX (5):

**FIG. 3 is an electrical block diagram including pertinent logical
equations
illustrating how the first level of logic of FIG. 1 may be employed in
conjunction with carry look-ahead and conditional selection logic
to provide a
high speed 4-bit conditional sum adder in accordance with the
invention.**

Detailed Description Text - DETX (9):

**Referring next to Fig. 3, illustrated therein is a 4-bit conditional
sum
adder in accordance with the invention which employs the 1-bit
conditional sum
adder and carry decode logic circuits of FIG. 1 as an input level. It
will be
seen in FIG. 3 that the carry propagate outputs P.sub.0-3 (that is
P.sub.0,
P.sub.1, P.sub.2 and P.sub.3) provided by the FIG. 1 portion are
applied to
both of carry look-ahead circuits logic 16 and 18, the carry
generate outputs
G.sub.0-3 (that is, G.sub.0, G.sub.1, G.sub.2 and G.sub.3) of FIG. 1
are
applied only to carry look-ahead logic circuit 16, and the carry
annihilate
outputs A.sub.0-3 (that is, A.sub.0, A.sub.1, A.sub.2 and A.sub.3)
are applied
only to carry look-ahead logic circuit 18. These carry look-ahead**

logic

circuits 16 and 18 are preferably identical and may comprise conventional carry look-ahead logic circuitry, such as is commercially available on a Fairchild CTL 9823 integrated circuit chip.

Detailed Description Text - DETX (10):

The carry look-ahead logic circuit 16 in FIG. 3 operates in response to an input carry C.sub.1 and the applied carry propagate and carry generate signals P.sub.0-3 and G.sub.0-3 to produce carry look-ahead signals C.sub.0, C.sub.1, C.sub.2 and C.sub.3, one for each bit position of the input operands. The carry look-ahead logic circuit 18, on the other hand, operates in response to the input carry C.sub.1 and the applied carry propagate and carry annihilate signals P.sub.0-3 and A.sub.0-3 to produce carry-not look-ahead signals C.sub.0, C.sub.1, C.sub.2, and C.sub.3. The logical equations for these carry and carry-not look-ahead signals are provided in FIG. 3 adjacent their respective output lines from the carry look-ahead logic circuits 16 and 18. These carry and carry-not look-ahead signals are applied to respective ones of AND gates 20-27 in FIG. 3 along with respective ones of the conditional sums S.sub.0-3 ' and S.sub.0-3 .degree. for selecting the conditional sums to be used in providing the proper resulting output sums S.sub.0,

**S.sub.1, S.sub.2
and S.sub.3 which properly represent the sum of the input
operands X.sub.0-3
and Y.sub.0-3.**

Detailed Description Text - DETX (11):

**It is to be noted that, in addition to the economy and high speed
made
possible by the combination of carry look-ahead and conditional
approaches
employed in FIG. 3, a further speed advantage is achieved by
providing the
carry annihilate terms A.sub.0-3 for application to the carry
look-ahead logic
circuit 18 along with the carry propagate signals P.sub.0-3 so that
the
carry-not look-ahead signals C.sub.0, C.sub.1, C.sub.2 and C.sub.3
are
available simultaneously with the carry look-ahead signals
C.sub.0, C.sub.1,
C.sub.2 and C.sub.3, rather than suffering the time delay which
would be
necessary if the carry-not look-ahead signals were to be derived
by inverting
the carry look-ahead signals C.sub.0, C.sub.1, C.sub.2 and C.sub.3.
It is also
to be noted that the logical equations for these carry-not signals
C.sub.0,
C.sub.1, C.sub.2 and C.sub.3 illustrated in FIG. 3 identically
correspond to
those of the carry look-ahead signals C.sub.0, C.sub.1, C.sub.2 and
C.sub.3,
except that the carry annihilate terms A.sub. 0, A.sub.1, A.sub.2
and A.sub.3
are substituted for respective ones of the carry generate terms**

**G.sub.0,
G.sub.1, G.sub.2 and G.sub.3, and the inverse carry-in signal
C.sub.1 is
substituted for the carry-in signal C.sub.1, thereby permitting the
same
circuits to be used for both of the carry look-ahead logic circuits
16 and 18.**

Detailed Description Text - DETX (12):

**It has thus been illustrated with reference to FIGS. 1-3 how the
basic
combined conditional and look-ahead approach of the invention
may be employed
to provide a high speed 4-bit conditional sum adder. It will be
apparent to
those skilled in the art that the same basic approach illustrated in
FIGS. 1-3
can readily be extended to handle any desired number of operand
bits by
increasing the number of stages in FIG. 1 and by providing the
required
increased complexity for the carry look-ahead logic circuits 16
and 18 in FIG.**

**3. However, it will be appreciated that the carry look-ahead
circuits will
become increasingly complex and expensive as the number of
operand bits
increases. Accordingly, another important feature of the present
invention
resides in the manner in which the basic combined conditional and
look-ahead
approach of the present invention, such as illustrated in
connection with FIGS.
1-3, can be economically extended to handle a relatively large
number of**

operand bits. For this purpose, attention is now directed to FIG. 4 which illustrates how a 4-bit conditional sum adder stage may be provided in accordance with the invention for producing two group conditional sums S_{g0-g3} and S'_{g0-g3} for the four least significant bits of the input operands along with group carry propagate, generate and annihilate signals P_{gl} , G_{gl} and A_{gl} . As illustrated in FIG. 5 to be considered hereinafter, these group conditional sums and group carry propagate, generate and annihilate signals produced by the stage of FIG. 4 may be employed along with those produced by a plurality of like stages to form a 16-bit high speed adder.

Detailed Description Text - DETX (13):

Now considering FIG. 4 in more detail, it will be seen that, as in FIG. 3, the circuit of FIG. 1 is again used as an input logic level for providing the conditional sums S_{0-3} and S'_{0-3} and the carry propagate, generate and annihilate terms P_{0-3} , G_{0-3} and A_{0-3} for the four least significant bit positions of the input operands, the higher order bits of the operands being fed to other like stages, as illustrated in FIG. 5. The stages 10-13 of FIG. 1 are thus sub-stages of the 4-bit adder stage of FIG. 4.

Four bits have been chosen for application to the 4-bit adder stage of FIG. 4, since such a choice permits use of the same relatively simple and inexpensive type of commercially available integrated circuit chip for the carry look-ahead logic circuits 30-33 in FIG. 4, as may be employed for the carry look-ahead logic circuits 16 and 18 in FIG. 3.

Detailed Description Text - DETX (14):

The embodiment of FIG. 4 employs the same basic combination of look-ahead and conditional circuitry used in FIG. 3. However, since two group conditional sums $S_{\text{sub}.g0-g3}^{\text{degree}}$ and $S_{\text{sub}.g0-g3}'$ are to be produced at the output in FIG. 4, one assuming a 0 carry-in and the other assuming a 1 carry-in, rather than the single sum $S_{\text{sub}.0-3}$ produced in FIG. 3, the embodiment of FIG. 4 employs two additional carry look-ahead logic circuits for a total of four (indicated by 30-33 in FIG. 4) in order to provide these two group conditional sums. Accordingly, carry look-ahead logic circuits 30 and 31 assume a 0 carry-in is present (that is, $C_{\text{sub}.l} = 0$ and $C_{\text{sub}.l} = 1$), while carry look-ahead logic circuits 32 and 33 assume that a 1 carry-in is present (that is, $C_{\text{sub}.l} = 1$ and $C_{\text{sub}.l} = 0$). Thus, carry look-ahead logic circuits 30 and 31 in FIG. 4 operate in response to respective ones of the carry propagate, generate and annihilate signals $P_{\text{sub}.0-3}$, $G_{\text{sub}.0-3}$, and $A_{\text{sub}.0-3}$

to produce
 conditional look-ahead carry and carry-not signals $C_{sub.0}^{degree}$
 , $C_{sub.1}^{degree}$
 $C_{sub.2}^{degree}$, $C_{sub.3}^{degree}$ and $C_{sub.0}^{degree}$
 ,
 $C_{sub.1}^{degree}$, $C_{sub.2}^{degree}$, $C_{sub.3}^{degree}$ which
 assume a 0
carry-in, while look-ahead carry logic circuits 32 and 33 operate to
 produce
 conditional look-ahead carry and carry-not signals $C_{sub.0}'$,
 $C_{sub.1}'$,
 $C_{sub.2}'$, $C_{sub.3}'$, and $C_{sub.0}'$, $C_{sub.1}'$, $C_{sub.2}'$, $C_{sub.3}'$
 which
 assume a 1 carry-in. As shown in FIG. 4 these conditional
 look-ahead carry and
 carry-not signals are applied to respective ones of AND gates
 35-46 along with
 respective ones of the conditional sums $S_{sub.0-3}^{degree}$ and $S_{sub.0-3}^{degree}$
 from FIG. 1 to provide the desired two group conditional sums
 $S_{sub.g0-g3}^{degree}$ and $S_{sub.g0-g3}'$ required for the stage. It will be
 understood that
 the logical equations indicated at the outputs of the look-ahead
 logic circuits
 30-33 in FIG. 4 for the look-ahead carry and carry-out signals are
 simplified
 as compared to those in FIG. 3 because the values of $C_{sub.1}$ and
 $C_{sub.1}$ are
 assumed to have the fixed values shown. As a result, since the
 values of
 $C_{sub.0}^{degree}$, $C_{sub.0}^{degree}$, $C_{sub.0}'$ and $C_{sub.0}'$ are
 known,
 there is no need to provide corresponding AND gates therefor, and
 they are
 omitted in FIG. 4.

Detailed Description Text - DETX (15):

The remaining signals required to be produced by the stage of FIG. 4 are the group carry propagate, generate and annihilate signals $P_{sub.gl}$, $G_{sub.gl}$ and $A_{sub.gl}$, the applicable logical equations for each being provided in FIG. 4 adjacent its respective output line. It is to be understood that the group carry generate signal $G_{sub.gl}$ and the group annihilate signal $A_{sub.gl}$ can readily be provided, since a Fairchild CTL 9823 logic chip, which as mentioned previously may be employed for each of the circuits 30-33, contains sufficient logic circuitry to additionally provide either $G_{sub.gl}$ or $A_{sub.gl}$ along with the required look-ahead carry output signals. Accordingly, the group generate and group annihilate signals $G_{sub.gl}$ and $A_{sub.gl}$ are illustrated in FIG. 4 as being provided by carry look-ahead logic circuits 30 and 34, respectively. In this regard, it is to be noted that the group carry annihilate signal $A_{sub.gl}$ can readily be provided by the same type of carry look-ahead logic circuit as is used to generate the group carry generate signal $G_{sub.gl}$, since the logical equation for the group carry annihilate signal $A_{sub.gl}$ is the same as that for the group carry generate signal $G_{sub.gl}$ except that the carry annihilate terms $A_{sub.0}$, $A_{sub.1}$, $A_{sub.2}$ and $A_{sub.3}$ are

substituted for
respective ones of the carry generate terms G.sub.0, G.sub.1,
G.sub.2 and
G.sub.3. It should thus now be apparent from FIG. 4 how the basic
combined
look-ahead and conditional approach illustrated in FIG. 3 can be
extended to
provide a 4-bit conditional sum adder stage. Reference is now
made to FIG. 5
which illustrates how the stage of FIG. 4 can be combined with
like stages to
form a high speed adder capable of handling operands having a
relatively large
number of bits.

Detailed Description Text - DETX (17):

Still with reference to FIG. 5, it will be apparent from a
comparison with
FIG. 3 that the 4-bit conditional sum adder stages 50-53 are
employed in FIG. 5
in place of the 1-bit adders 10-13 of FIG. 1. It will also be apparent
that
the carry look-ahead logic circuits 52 and 54 in FIG. 5 provide the
same
logical manipulations on respective ones of the group carry
propagate, generate
and annihilate signals P.sub.g1-gIV, G.sub.g1-gIV and A.sub.g1-gIV
for
producing the group carry look-ahead signals C.sub.g0, C.sub.g1,
C.sub.g2 and
C.sub.g3 and the group carry-not look-ahead signals C.sub.g0,
C.sub.g1,
C.sub.g2 and C.sub.g3 as do the carry look-ahead logic circuits 16
and 18 in
FIG. 3 in producing the carry look-ahead signals C.sub.0, C.sub.1,

**C.sub.2 and
C.sub.3 and the carry-not look-ahead signals C.sub.0, C.sub.1,
C.sub.2 and
C.sub.3. Thus, the same commercially available Fairchild CTL
9823 chip can be
used for each of the carry look-ahead logic circuits 52 and 54 as
well as for
the circuits 16 and 18 in FIG. 3 and the circuits 30-33 in FIG. 4.**

Detailed Description Text - DETX (18):

**In the same basic manner as in FIG. 3, the resulting group
look-ahead carry
and group look-ahead carry-not signals C.sub.g0, C.sub.g1,
C.sub.g2, C.sub.g3
and C.sub.g0, C.sub.g1, C.sub.g2 and C.sub.g3 in Fig. 5 are applied
to
respective groups of AND gates 60-67 along with respective ones
of the group
conditional sums S.sub.g0-g3', S.sub.g4-g7', S.sub.g8-g11',
S.sub.g12-g15' and
S.sub.g0-g3.degree., S.sub.g4-g7.degree., S.sub.g8-g11.degree.,
S.sub.g12-g15.degree. so as to permit appropriate selection of the
group
conditional sums as required to provide resulting sums S.sub.0-3,
S.sub.4-7,
S.sub.8-11 and S.sub.12-15 which will properly represent the sum
of the input
operands X.sub.0-3, Y.sub.4-7, Y.sub.8-11, X.sub.12-15 and
Y.sub.0-3,
Y.sub.4-7, Y.sub.8-11 and Y.sub.12-15.**

Detailed Description Text - DETX (20):

**As pointed out previously, it is advantageous to employ the
combined**

look-ahead and conditional approach of the invention, not only for the larger adder of FIG. 5, but also for each of the adder stages 50-53 using the adder stage embodiment illustrated in FIG. 4. However, it is to be understood that the adder stages 50-53 in FIG. 5 need not be designed in this manner, in order to take advantage of the FIG. 5 approach. For example, an alternate design for a 4-bit conditional sum adder stage is illustrated in FIG. 6 which is somewhat more economical as regards circuit implementation, but which is not as fast. Considering the FIG. 6 embodiment in more detail, it will be noted that the conditional sums S_{0-3} and S'_{0-3} are not provided in an input logic level along with the carry propagate, generate and annihilate terms P_{0-3} , G_{0-3} and A_{0-3} as is done in FIG. 4. Also, the FIG. 6 embodiment does not form conditional look-ahead carry-not signals as is done in FIG. 4 using the annihilate signals A_{0-3} in conjunction with the two additional look-ahead carry logic circuits 31 and 33. Instead, in the FIG. 6 embodiment, the four least significant bits X_{0-3} and Y_{0-3} of the input operands are applied to a conventional form of 4-bit carry decode logic circuit 70 which produces only the carry propagate, generate and annihilate signals P_{0-3} , G_{0-3} and A_{0-3} , with an ADD/SUBTRACT control also

being provided as in Fig. 4 to permit the performance of either addition or subtraction. The carry propagate and generate signals $P_{sub.0-3}$ and $G_{sub.0-3}$ are then applied to two carry look-ahead logic circuits 72 and 74 for generating the conditional carry look-ahead signals $C_{sub.0.degree.}$, $C_{sub.1.degree.}$, $C_{sub.2.degree.}$, $C_{sub.3.degree.}$ and $C_{sub.0'}$, $C_{sub.1'}$, $C_{sub.2'}$, $C_{sub.3'}$ which are in turn applied to respective stages of two conventional 4-bit full adders 82 and 84 along with respective ones of the input operand bits $x_{sub.0-3}$ and $Y_{sub.0-3}$ to thereby provide the group conditional sums $S_{sub.g0-g3.degree.}$ and $S_{sub.g0-g3'}$. In the FIG. 4 embodiment these sums were provided using the selection AND gates 20-27.

Detailed Description Text - DETX (21):

As in FIG. 4, the group propagate term $P_{sub.g1}$ in FIG. 6 is derived from the carry propagate signals $P_{sub.0-3}$ using an AND gate 76, while the group generate signal $G_{sub.g1}$ is derived from one of the carry look-ahead logic circuits 74. The group annihilate signal $A_{sub.g1}$ in FIG. 6 is derived from the propagate and annihilate signals $P_{sub.0-3}$ and $A_{sub.0-3}$ using appropriate group annihilate logic 78 to thereby complete the signals required for the adder stage. It will be understood that the carry look-ahead logic

**circuits 72
and 74 in FIG. 6 may each be implemented using the same
previously referred to
Fairchild CTL 9823 integrated circuit chip. The group annihilate
logic 78 may
also be implemented using this same chip, while the 4-bit carry
decode logic 70
as well as the 4-bit full adders 82 may be implemented using the
same Fairchild
CTL 9838 integrated circuit chips used for implementing the
stages of FIG. 1.**

Detailed Description Text - DETX (23):

**Referring next to FIG. 7, illustrated therein is still another
advantageous
embodiment in accordance with the invention for providing for the
handling of
large numbers of operand bits. FIG. 7 illustrates how a 16-bit
adder stage can
be provided which can be combined with three other like 16-bit
adder stages, in
the same manner as illustrated for the 4-bit stages in FIG. 5, so as
to thereby
provide a 64-bit adder. In other words, the 4-bit conditional sum
adder stage
50-53 of FIG. 5 are now sub-stages of the 16-bit conditional sum
adder stage of
Fig. 7. A particular advantage of this approach is that the resulting
64-bit
adder can be provided using the same relatively simple and
inexpensive
Fairchild CTL 9823 integrated circuit chip for each carry
look-ahead circuit
because of the modular approach employed.**

Detailed Description Text - DETX (24):

More specifically, as illustrated in Fig. 7, it will be seen that the same 4-bit conditional sum adder stages 50-53 employed in the 16-bit adder of FIG. 5 are also used in the embodiment of FIG. 7 for providing the group conditional sums $S_{\text{sub.g0-g15.degree.}}$ and $S_{\text{sub.g0-g15'}}$ and the group carry propagate, generate and annihilate signals $P_{\text{sub.g1-g1V}}$, $G_{\text{sub.g1-g1V}}$, and $A_{\text{sub.g1-g1V}}$ for the sixteen least significant digits $X_{\text{sub.0-15}}$ and $Y_{\text{sub.0-15}}$ of a pair of 64-bit input operands. These group conditional sums $S_{\text{sub.g0-g15.degree.}}$ and $S_{\text{sub.g0-g15'}}$ and the group carry propagate, generate and annihilate signals $P_{\text{sub.g1-g1V}}$, $G_{\text{sub.g1-g1V}}$ and $A_{\text{sub.g1-g1V}}$ are applied to carry look-ahead logic circuits 90-93 in FIG. 7 (which perform the same logical operations thereon as do the like carry look-ahead logic circuits 30-33 in FIG. 4 on the carry propagate, generate and annihilate signals $P_{\text{sub.0-3}}$, $G_{\text{sub.0-3}}$ and $A_{\text{sub.0-3}}$) to provide group conditional look-ahead carry signals $gC_{\text{sub.g0.degree.}}$, $gC_{\text{sub.g1.degree.}}$, $gC_{\text{sub.g2.degree.}}$, $gC_{\text{sub.g3.degree.}}$ and $gC_{\text{sub.g0'}}$, $gC_{\text{sub.g1'}}$, $gC_{\text{sub.g2'}}$, $gC_{\text{sub.g3'}}$ and group conditional look-ahead carry-not signals $gC_{\text{sub.g0.degree.}}$, $gC_{\text{sub.g1.degree.}}$, $gC_{\text{sub.g2.degree.}}$, $gC_{\text{sub.g3.degree.}}$, and $gC_{\text{sub.g0'}}$, $gC_{\text{sub.g1'}}$

g2'gC.sub.g2 ' , gC.sub.g3 ' . As illustrated in FIG. 7, these conditional look-ahead carry and carry-not signals are applied to respective & gate groups 100-111 along with respective ones of the group conditional sums S.sub.g0-g15 ' and S.sub.g0-g15 .degree. for providing the resulting group conditional sums gS.sub.g0-g15 .degree. and gS.sub.g0-g15 ' required for the 16-bit adder stage. It will thus be understood that a 64-bit adder can now readily be provided by merely substituting the 16-bit adder stage of FIG. 7 for each 4-bit adder stage of FIG. 5, and by providing sufficient numbers of AND gates in the & gates groups 60-67 in FIG. 5, one for each bit position, so as to thereby provide for selection of the appropriate conditional sums required for the resulting sum of the 64-bit input operands.

Claims Text - CLTX (3):

carry look-ahead logic means responsive to said carry output signals for producing stage look-ahead carry and carry-not signals; and

Claims Text - CLTX (8):

wherein said carry look-ahead logic means comprises a first carry look-ahead logic circuit responsive to said carry propagate and carry generate signals for producing said carry look-ahead signals, and a second carry look-ahead logic

circuit providing the same logical manipulations as said first carry look-ahead logic circuit and responsive to said carry propagate and carry annihilate signals for producing said look-ahead carry-not signals.

Claims Text - CLTX (12):

wherein said carry look-ahead logic means comprises a first carry look-ahead logic means responsive to said carry propagate and carry generate signals for producing said carry look-ahead signals, and a second carry look-ahead logic means responsive to said carry propagate and carry annihilate signals for producing said look-ahead carry-not signals.

Claims Text - CLTX (13):

5. The invention in accordance with claim 4, wherein said first carry look-ahead logic means is additionally responsive to a carry-in signal provided along with said operands, while said second carry look-ahead logic means is additionally responsive to the inverse of said carry-in signal.

Claims Text - CLTX (16):

carry look-ahead logic means for each stage responsive to the carry output signals produced by the sub-stages for producing first and second sub-stage look-ahead carry and carry-not signals assuming the presence and absence,

respectively, of a carry-in to the stage;

Claims Text - CLTX (23):

wherein said carry look-ahead logic means responsive to said stage carry output signals comprises a first carry look-ahead logic circuit responsive to said stage carry propagate and generate signals for producing said stage carry look-ahead signals and a second carry logic circuit responsive to said stage carry propagate and annihilate signals for producing said stage carry-not look-ahead signals; and

Claims Text - CLTX (24):

wherein said carry look-ahead logic means responsive to said sub-stage carry output signals comprises third, fourth, fifth and sixth carry look-ahead logic circuits, said third and fourth carry look-ahead logic circuits being responsive to said sub-stage carry propagate and generate signals and assuming the presence and absence, respectively, of a carry-in to the stage for producing said first and second sub-stage look-ahead carry signals, and said fifth and sixth carry look-ahead logic circuits being responsive to said sub-stage carry propagate and annihilate signals and assuming the presence and absence, respectively, of a carry-in to the stage for producing said first and second sub-stage look-ahead carry-not signals.

Claims Text - CLTX (25):

9. The invention in accordance with claim 8, wherein said first carry look-ahead logic circuit is additionally responsive to a carry-in signal provided along with said operands while said second carry look-ahead logic circuit is additionally responsive to the inverse of said carry-in signal.

Claims Text - CLTX (26):

10. The invention in accordance with claim 9 wherein all of said carry look-ahead logic circuits employ the same logical circuitry.

Claims Text - CLTX (27):

11. The invention in accordance with claim 1, wherein each stage receives a predetermined plurality of different pairs of consecutive corresponding digits of said operands, wherein said conditional sum signals for each stage comprise a first group conditional sum signal which assumes the presence of a carry-in to the stage and a second group conditional sum signal which assumes the absence of a carry-in to the stage, wherein said carry information output signals produced for each stage comprise a group carry propagate signal, a group carry generate signal and a group carry annihilate signal, and wherein

said carry look-ahead logic means comprises first carry look-ahead logic means responsive to said group carry propagate and group carry generate signals for producing said look-ahead carry signals and a second carry look-ahead logic means responsive to said group carry propagate and said group carry annihilate signals for producing said look-ahead carry-not signals.

Claims Text - CLTX (28):

12. The invention in accordance with claim 11, wherein said first carry look-ahead logic means is additionally responsive to a carry-in signal provided along with said operands, while said second carry look-ahead logic means is additionally responsive to the inverse of said carry-in signal.

Claims Text - CLTX (31):

first, second, third and fourth carry look-ahead logic means for each stage responsive to predetermined ones of said sub-stage carry signals for producing first and second sub-stage look-ahead carry and carry-not signals assuming the presence and absence, respectively, of a carry-in to the stage;

Claims Text - CLTX (36):

first carry look-ahead logic means responsive to said sub-stage carry propagate and generate signals for producing first look-ahead carry-in signals

which assume the presence of a carry-in to the stage;

Claims Text - CLTX (37):

second carry look-ahead logic means responsive to said sub-stage carry propagate and generate signals for producing second look-ahead carry-in signals which assume the absence of a carry-in to the stage;

Claims Text - CLTX (43):

carry look-ahead logic means responsive to the carry output signals produced by the sub-stages for producing first and second sub-stage look-ahead carry and carry-not signals assuming the presence and absence, respectively, of a carry-in to the stage;

Claims Text - CLTX (47):

wherein said carry look-ahead logic means responsive to said sub-stage carry output signals comprises third, fourth, fifth and sixth carry look-ahead logic circuits, said third and fourth carry look-ahead logic circuits being responsive to said sub-stage carry propagate and generate signals and assuming the presence and absence, respectively, of a carry-in to the stage for producing said first and second sub-stage look-ahead carry signals, and said fifth and sixth carry look-ahead logic circuits being responsive to said sub-stage carry propagate and annihilate signals and assuming the

presence and
absence, respectively, of a carry-in to the stage for producing said
first and
second sub-stage look-ahead carry-not signals.

Claims Text - CLTX (50):

electronically combining said carry output signals for producing
look-ahead
carry and carry-not signals; and

Claims Text - CLTX (54):

wherein the step of electronically combining comprises
generating said
look-ahead carry signals in response to said carry propagate and
generate
signals, and generating said look-ahead carry-not signals in
response to said
carry propagate and carry annihilate signals.

US-PAT-NO: 5964827

DOCUMENT-IDENTIFIER: US 5964827 A

TITLE: High-speed binary adder

DATE-ISSUED: October 12, 1999

US-CL-CURRENT: 708/710

APPL-NO: 08/ 971653

DATE FILED: November 17, 1997

----- KWIC -----

Claims Text - CLTX (2):

a plurality of rows of carry-lookahead circuits, wherein a first row of said plurality of rows includes a plurality of four-bit group generate circuits and a plurality of four-bit group propagate circuits, wherein each of said four-bit group generate circuits logically combines bits from four different bit positions within said multiple-bit addends to generate a group generate signal, wherein each of said four-bit group propagate circuits logically combines bits from four different bit positions within said multiple-bit addends to generate

a group propagate signal;

Claims Text - CLTX (17):

a plurality of rows of carry-lookahead circuits, wherein a first row of said plurality of rows includes a plurality of four-bit group generate circuits and a plurality of four-bit group propagate circuits, wherein each of said four-bit group generate circuits logically combines bits from four different bit positions within said multiple-bit addends to generate a group generate signal, wherein each of said four-bit group propagate circuits logically combines bits from four different bit positions within said multiple-bit addends to generate a group propagate signal;

Other Reference Publication - OREF (2):

"Improved Zero Result Detection When Using a Carry Look-Ahead Adder," IBM Technical Disclosure Bulletin, vol. 30, No. 11, Apr. 1988, pp. 288-290.

US-PAT-NO: 5047974

DOCUMENT-IDENTIFIER: US 5047974 A

**TITLE: Cell based adder with tree structured carry,
inverting
logic and balanced loading**

DATE-ISSUED: September 10, 1991

US-CL-CURRENT: 708/712

DISCLAIMER DATE: 20061121

APPL-NO: 07/ 124807

DATE FILED: November 24, 1987

----- KWIC -----

Brief Summary Text - BSTX (25):

This implementation generally requires the most hardware but give the fastest results because the delay grows as $\log_{2.5} n$ rather than being proportional to n as in ripple carry and look ahead carry. It should be noted that the constants for ripple carry, look ahead carry and the structured carry are not necessarily the same. A tree structure adder generates all propagate

and generate terms in parallel then combines the propagate and generate terms of bit position N with lower order bits to form the complete carry term
C.sub.N.

Claims Text - CLTX (4):

each of said output portions being coupled to said carry propagation portions and to said front end portions, and including means for logically combining propagate and generate signals with carry propagation signals to derive a summation output code representative of the sum of said first and second digital code signals.

Claims Text - CLTX (8):

5. An adder according to claim 1, wherein the carry propagation portion of a respective adder cell comprises a first NOR gate having first and second inputs coupled to receive respective ones of said propagate signals and a first OR/NAND gate having first and second inputs coupled to receive respective ones of said generate signals and a third input coupled to one of the inputs of said first NOR gate, the outputs of said first NOR gate and said first OR/NAND gate being coupled to logically combining means of selected ones of said output portions.

Claims Text - CLTX (9):

6. An adder according to claim 5, wherein said logically combining means of a respective output portion includes a second OR/NAND gate having first and second inputs coupled to receive propagate and generate signals from selected ones of said front end portions, a third input coupled to receive a carry signal, and an output, and a first exclusive NOR gate having a first input coupled to the output of said second OR/NAND gate and a second input coupled to a front end portion, and an output from which a SUM term signal is derived.

Claims Text - CLTX (10):

7. An adder according to claim 5, wherein said logically combining means of a respective output portion includes a second OR/NAND gate having first and second inputs coupled to receive propagate and generate signals from selected ones of said front end portions, a third input coupled to receive a carry signal, and an output, and a first exclusive OR gate having a first input coupled to the output of said second OR/NAND gate and a second input coupled to a front end portion, and an output from which a SUM term signal is derived.

Claims Text - CLTX (14):

an output portion, coupled to said carry propagation portion and to said front end portion, and including means for logically combining propagate and generate signals with carry propagation signals to derive a summation output code representative of the sum of said first and second digital code signals.

Claims Text - CLTX (61):

an output portion, coupled to said carry propagation portion and to said front end portion, and including means for logically combining propagate and generate signals with carry propagation signals to derive summation output bit values representative of the sum of the respective bit values for said bits positions of said first and second digital code signals;

US-PAT-NO: 4584661

DOCUMENT-IDENTIFIER: US 4584661 A

**TITLE: Multi-bit arithmetic logic units having fast parallel
carry systems**

DATE-ISSUED: April 22, 1986

US-CL-CURRENT: 708/712

APPL-NO: 06/ 659512

DATE FILED: October 11, 1984

PARENT-CASE:

This application is a continuation of Ser. No. 503,062, filed June 10, 1983, now abandoned, which is in turn a continuation of application Ser. No. 256,405, filed Apr. 22, 1981, now abandoned.

FOREIGN-APPL-PRIORITY-DATA:

COUNTRY	APPL-NO	APPL-DATE
IL	59907	April 23, 1980

----- KWIC -----

Abstract Text - ABTX (1):

A multi-bit arithmetic logic unit having a fast carry-look-ahead

(CLA)

system comprises logical preparation (LLP) circuitry of the first CLA level

which interconnects several bit positions so as to output intermediate

combination signals which are a function of a plurality of bit positions,

rather than of a single bit position. In addition, the carry-in signal (C.sub.IN) into the least-significant-bit (LSB) of the respective group is

separated from the intermediate combined propagated carry signal (C.sub.p)

between groups, and is introduced into the input of the LLP circuitry of the

first CLA level, combined with the LSB signals, and also with a binary "split"

signal whose state determines whether the C.sub.IN is to be inputted or not.

In addition, the ALU may be comprised of a plurality of identical integrated

circuit chips called Y-chips, each including LLP and logical combination (LLC)

circuitries of two CLA levels having reverse input-output logic such that a

plurality of such chips may be connected together to form a two-level CLA, and

may also be connected together, with the addition of further chips, called

Z-chips, having a one-level CLA, to produce three-level, or higher-level, CLA

systems.

Brief Summary Text - BSTX (2):

The present invention relates to multibit arithmetic logic units (ALU's),

more specifically, multibit adders-subtractors (A/S), having fast carry-look-ahead (CLA) systems, and particularly to a CLA system designed to speed-up the execution of arithmetical operations.

Brief Summary Text - BSTX (11):

The present invention is directed to ALU's including a carry-look-ahead (CLA) system having a plurality of CLA levels, each level including;

(1)

logical preparation (LLP) circuitry in which are produced intermediate combined signals, the generated carries (G.sub.N) and the propagated carries (P.sub.N) of the respective bit position (A.sub.N, B.sub.N) and G.sub.0, P.sub.0, for the respective group or section position; and (2) logical combination (LLC) circuitry in which are produced the propagated combined carry signals, i.e. the final combined propagated carry (C.sub.N), for the first CLA level, or the intermediate combined propagated carry signals (C.sub.P) for other than the first CLA level.

Detailed Description Text - DETX (3):

As known, the arithmetic portion of an ALU is divided into two parts, namely, the Sum Part and the Carry-Look-Ahead (CLA)Part. The CLA part includes several CLA levels in order to reduce propagation delay, which thereby speeds up the execution of arithmetic functions. The total number of bits

per ALU
 (N.sub.F) executing 2's complement operations is equal to the
 total number (N)
 of bits per group, group per sections, sections per division, etc.
 raised to
 the power of the number of participating CLA levels (L); thus:

$$N_{sub.F} = N_{sup.L}$$

Claims Text - CLTX (1):

1. An arithmetic-logic-unit (ALU) comprising a plurality of bits
 arranged
 according to at least groups and sections including a
carry-look-ahead (CLA)
 system having a plurality of CLA levels, each CLA level including
 logical
 preparation (LLP) circuitry in which are produced the intermediate
combined
signals, the generated carries (G.sub.N) and the propagated
carries (P.sub.N)
 of the respective bit position (A.sub.N, B.sub.N), (P.sub.o, G.sub.o)
 for the
 respective group or section position and the logical combination
 (LLC)
 circuitry in which are produced the final combined propagated
carry signals
 (C.sub.N) for the first CLA level, or the intermediate combined
propagated
carry signals (C.sub.P) for the other CLA levels; said LLP circuitry
 of the
 first CLA level interconnecting several bit positions so as to output
 from said
 first-level LLP circuitry the intermediate combination signals which
 are a
 function of a plurality of bit positions, rather than of a single bit

position,
thus expanding the number of bits per group while drastically
reducing extreme
fan-out loading expanding the number of groups per section, and
maintaining the
overflow signal propagation delay in-phase with the entire adder.

Claims Text - CLTX (5):

5. An arithmetic logic unit (ALU) including a plural-level carry-look-ahead (CLA) system having logical preparation (LLP) circuitry in which are produced intermediate combined signals, the generated carries (G.sub.N), and the propagated carries (P.sub.N) of the respective position (A.sub.N B.sub.N), and logical combination (LLC) circuitry in which are produced the propagated combined carry signals for the respective CLA level; said ALU including means for separating the carry-in signal (C.sub.IN) into the least significant-bit (LSB) of the respective group, from the intermediate combined propagated carry signal (C.sub.P) between groups, and for introducing same into the input of the first-level LLP circuitry combined with the LSB signals, whereas the C.sub.P signal is introduced into the LLC circuitry input of the respective CLA level.

Claims Text - CLTX (11):

levels of sum parts and carry-look-ahead (CLA) parts, each CLA level

including logical preparation (LLP) circuitry in which are produced intermediate combined signals, the generated carries (G.sub.N), and the propagated carries (P.sub.N) of the respective bit position (A.sub.N, B.sub.N), (P.sub.o, G.sub.o) for the respective group or section position, and the logical combination (LLC) circuitry in which are produced the propagated combined carry signals C.sub.N for the first CLA level, or the intermediate combined propagated carry signals (C.sub.P) for the other CLA levels; said ALU incorporating split means (SP) for splitting said adder-subtractor between said groups and sections so as to form therefrom at least two independent same-sense 2's compliment adders/subtractors, thus enabling, on the one hand, simultaneous multiple parallel processing of data having variable data length, and on the other hand, selecting the degree of accuracy desired.

Claims Text - CLTX (14):

13. An arithmetic logic unit (ALU) including a plural-level parallel carry system having logical preparation (LLP) circuitry in which are produced intermediate combined signals, the generated carries (G.sub.N) and the propagated carries (P.sub.N) of the respective position (A.sub.N, B.sub.N); and logical combination (LLC) circuitry in which are produced the propagated combined carry signals for the respective carry level; said ALU

including
input means for introducing the carry-in (C.sub.IN) signal into the
immediate
first-level LLP circuitry combined with the least-significant-bit LSB
signals
"A.sub.1, B.sub.1 " and an optional split signal (SP), the mentioned
signals
having the following relationships: ##EQU10##